CSE508 Information Retrieval

Winter 2023
Assignment-2
Group 70

Vedant Patil(2020348) Ashish Kamathi(2020364) Ashwin Sheoran(2020288)

Ans 1)

First we did the pre processing

(i) Relevant Text Extraction

1st Printed the initial 5 files before and preprocessing.
We will also unpack the compressed file to get the folder of our dataset to use it.
Then loaded all the files one by one using BeautifulSoup.Then we separated the title and the text part of the files and concatenated only the title and text part of the files. Then we again wrote this concatenated string back to the original files and then printed the first 5 files.

(ii) Preprocessing

We loaded the dataset obtained after Title and Text extraction, Using the nltk library, we 1st lowercase the text.
Then we did tokenization, in which we also considered blank spaces as individual tokens by replacing them with '@' and after tokenization replacing them back to blank space, so they don't get deleted during tokenization.
We also considered a-b-c as three different tokens 'a' , 'b' , 'c'.
Then we removed stop words from the tokens using nltk library.
Then we removed punctuations using isalpha() function
Then we removed blank spaces by removing ' ' from tokens.
We then saved our token in the respective original files as string of list of tokens.

The result after preprocessing is shown below

```
case-1  Tokens are
['experimental', ' ', 'investigation', ' ', 'of', ' ', 'the', ' ', 'aerodynamics', ' ', 'of', ' ', 'a', 'wing', ' ', 'in', ' ', 'a', ' ', 'slipstream', ' ', '.', ' ', ' ', 'an', ' ', 'exp
erimental', ' ', 'study', ' ', 'of', ' ', 'a', ' ', 'wing', ' ', 'in', ' ', 'a', ' ', 'propeller', ' ', 'slipstream', ' ', 'was', 'made', ' ', 'in', ' ', 'order', ' ', 'to', ' ', 'determi
ne', ' ', 'the', ' ', 'spanwise', ' ', 'distribution', ' ', 'of', ' ', 'the', ' ', 'lift', 'increase', ' ', 'due', ' ', 'to', ' ', 'slipstream', ' ', 'at', ' ', 'different', ' ', 'angles
', ' ', 'of', ' ', 'attack', ' ', 'of', ' ', 'the', ' ', 'wing', 'and', ' ', 'at', ' ', 'different', ' ', 'free', ' ', 'stream', ' ', 'to', ' ', 'slipstream', ' ', 'velocity', ' ', 'ratios
', ' ', '.', ' ', ' ', 'the', 'results', ' ', 'were', ' ', 'intended', ' ', 'in', ' ', 'part', ' ', 'as', ' ', 'an', ' ', 'evaluation', ' ', 'basis', ' ', 'for', ' ', 'different', 'theore
tical', ' ', 'treatments', ' ', 'of', ' ', 'this', ' ', 'problem', ' ', '.', ' ', ' ', ' ', 'the', ' ', 'comparative', ' ', 'span', ' ', 'loading', ' ', 'curves', ',', ' ', 'together', ' ', 'w
ith', ' ', 'supporting', 'evidence', ',', ' ', 'showed', ' ', 'that', ' ', 'a', ' ', 'substantial', ' ', 'part', ' ', 'of', ' ', 'the', ' ', 'lift', ' ', 'increment', 'produced', ' ', 'by
', ' ', 'the', ' ', 'slipstream', ' ', 'was', ' ', 'due', ' ', 'to', ' ', 'a', ' ', '/destalling/', ' ', 'or', ' ', 'boundary', 'layer', 'control', 'effect', ' ', '.', ' ', ' ', ' ', 'the', '
', 'integrated', ' ', 'remaining', ' ', 'lift', ' ', 'increment', ',', 'after', ' ', 'subtracting', ' ', 'this', ' ', 'destalling', ' ', 'lift', ',', ' ', 'was', ' ', 'found', ' ', 'to',
' ', 'agree', 'well', ' ', 'with', ' ', 'a', ' ', 'potential', ' ', 'flow', ' ', 'theory', ' ', '.', ' ', ' ', 'an', ' ', 'empirical', ' ', 'evaluation', ' ', 'of', ' ', 'the', ' ', 'dest
alling', ' ', 'effects', ' ', 'was', ' ', 'made', ' ', 'for', 'the', ' ', 'specific', ' ', 'configuration', ' ', 'of', ' ', 'the', ' ', 'experiment', ' ', '.']


case-1  Tokens after removing stop Words
['experimental', ' ', 'investigation', ' ', ' ', ' ', ' ', 'aerodynamics', ' ', ' ', 'wing', ' ', ' ', ' ', 'slipstream', ' ', '.', ' ', ' ', ' ', ' ', 'experimental', ' ', 'study', ' ', ' ', ' ',
'wing', ' ', ' ', ' ', ' ', 'propeller', ' ', 'slipstream', ' ', 'made', ' ', ' ', 'order', ' ', ' ', ' ', 'determine', ' ', ' ', ' ', 'spanwise', ' ', 'distribution', ' ', ' ', ' ', ' ', 'lift', 'increase',
' ', 'due', ' ', ' ', 'slipstream', ' ', ' ', 'different', ' ', 'angles', ' ', ' ', ' ', 'attack', ' ', ' ', ' ', ' ', 'wing', ' ', ' ', ' ', 'different', ' ', 'free', ' ', 'stream', ' ', ' ', 'slipstr
eam', ' ', 'velocity', ' ', 'ratios', ' ', ' ', '.', ' ', ' ', 'results', ' ', ' ', ' ', 'intended', ' ', ' ', ' ', 'part', ' ', ' ', ' ', ' ', 'evaluation', ' ', 'basis', ' ', ' ', ' ', 'different', 'theoretical
', ' ', 'treatments', ' ', ' ', ' ', ' ', 'problem', ' ', '.', ' ', ' ', ' ', ' ', ' ', 'comparative', ' ', 'span', ' ', 'loading', ' ', 'curves', ',', ' ', 'together', ' ', ' ', ' ', 'supporting', 'evidence
', ',', ' ', ' ', 'showed', ' ', ' ', ' ', ' ', ' ', 'substantial', ' ', 'part', ' ', ' ', ' ', ' ', ' ', 'lift', ' ', ' ', 'increment', 'produced', ' ', ' ', ' ', ' ', 'slipstream', ' ', ' ', ' ', 'due', ' ', ' ', ' ', ' ', ' ', '/desta
lling/', ' ', ' ', 'boundary', 'layer', 'control', 'effect', ' ', '.', ' ', ' ', ' ', ' ', ' ', 'integrated', ' ', 'remaining', ' ', ' ', 'lift', ' ', 'increment', ',', ' ', 'subtracting', ' ', ' ', ' ', 'de
stalling', ' ', 'lift', ',', ' ', ' ', ' ', 'found', ' ', ' ', 'agree', 'well', ' ', ' ', ' ', ' ', 'potential', ' ', 'flow', ' ', 'theory', ' ', ' ', '.', ' ', ' ', ' ', ' ', 'empirical', ' ', 'evaluation',
' ', ' ', ' ', ' ', 'destalling', ' ', 'effects', ' ', ' ', ' ', 'made', ' ', ' ', 'specific', ' ', 'configuration', ' ', ' ', ' ', ' ', 'experiment', ' ', '.']


case-1  Tokens after removing punctiations
['experimental', ' ', 'investigation', ' ', ' ', ' ', ' ', 'aerodynamics', ' ', ' ', 'wing', ' ', ' ', ' ', ' ', 'slipstream', ' ', ' ', ' ', ' ', ' ', ' ', 'experimental', ' ', 'study', ' ', ' ', ' ', ' ', 'wing
', ' ', ' ', ' ', ' ', 'propeller', ' ', 'slipstream', ' ', 'made', ' ', ' ', ' ', 'order', ' ', ' ', ' ', 'determine', ' ', ' ', ' ', 'spanwise', ' ', 'distribution', ' ', ' ', ' ', ' ', 'lift', 'increase', ' ',
 'due', ' ', ' ', 'slipstream', ' ', ' ', ' ', 'different', ' ', 'angles', ' ', ' ', ' ', 'attack', ' ', ' ', ' ', ' ', 'wing', ' ', ' ', ' ', 'different', ' ', 'free', ' ', 'stream', ' ', ' ', ' ', 'slipstream',
 ' ', 'velocity', ' ', 'ratios', ' ', ' ', ' ', ' ', 'results', ' ', ' ', ' ', 'intended', ' ', ' ', ' ', 'part', ' ', ' ', ' ', ' ', 'evaluation', ' ', 'basis', ' ', ' ', ' ', 'different', 'theoretical', ' ', 't
reatments', ' ', ' ', ' ', ' ', 'problem', ' ', ' ', ' ', ' ', ' ', ' ', ' ', 'comparative', ' ', 'span', ' ', 'loading', ' ', ' ', 'curves', ' ', ' ', 'together', ' ', ' ', ' ', 'supporting', 'evidence', ' ', 'showed', ' '
', ' ', ' ', ' ', 'substantial', ' ', 'part', ' ', ' ', ' ', ' ', ' ', 'lift', ' ', ' ', 'increment', 'produced', ' ', ' ', ' ', ' ', 'slipstream', ' ', ' ', ' ', 'due', ' ', ' ', ' ', ' ', ' ', ' ', 'boundary', 'layer', '
control', 'effect', ' ', ' ', ' ', ' ', ' ', ' ', 'integrated', ' ', 'remaining', ' ', ' ', 'lift', ' ', ' ', 'increment', ' ', 'subtracting', ' ', ' ', ' ', 'destalling', ' ', 'lift', ' ', ' ', ' ', ' ', 'found', ' ', ' ', ' '
, 'agree', 'well', ' ', ' ', ' ', ' ', 'potential', ' ', 'flow', ' ', 'theory', ' ', ' ', ' ', ' ', ' ', 'empirical', ' ', 'evaluation', ' ', ' ', ' ', ' ', 'destalling', ' ', 'effects', ' ', ' ', ' ', 'made
', ' ', ' ', 'specific', ' ', 'configuration', ' ', ' ', ' ', ' ', 'experiment', ' ']


case-1 Tokens after removing Blank Spaces
['experimental', 'investigation', 'aerodynamics', 'wing', 'slipstream', 'experimental', 'study', 'wing', 'propeller', 'slipstream', 'made', 'order', 'determine', 'spanwise', 'distribution
', 'lift', 'increase', 'due', 'slipstream', 'different', 'angles', 'attack', 'wing', 'different', 'free', 'stream', 'slipstream', 'velocity', 'ratios', 'results', 'intended', 'part', 'eva
luation', 'basis', 'different', 'theoretical', 'treatments', 'problem', 'comparative', 'span', 'loading', 'curves', 'together', 'supporting', 'evidence', 'showed', 'substantial', 'part',
'lift', 'increment', 'produced', 'slipstream', 'due', 'boundary', 'layer', 'control', 'effect', 'integrated', 'remaining', 'lift', 'increment', 'subtracting', 'destalling', 'lift', 'found
', 'agree', 'well', 'potential', 'flow', 'theory', 'empirical', 'evaluation', 'destalling', 'effects', 'made', 'specific', 'configuration', 'experiment']
```

Then we created a vocab matrix having all the unique words. They we created functions for each of the different Weighting schemes using their given formula. Then we used these functions to create different tf-idf matrices. Then we created a query vector using an example query. Then we used this query vector to calculate the tf-idf scores of tf-idf matrices of different weighing schemes.

The scores are coming in as follows, where I have taken an example query “experimental investigation.

```
Query: experimental investigation
Top 5 relevant documents for Binary:
Document 1: experimental investigation aerodynamics wing slipstream experimental study wing propeller slipstream made order determine spanwise distribution lift increase d
 (score: 3.10)

Document 19: investigation pressure distribution conical bodies hypersonic flows large amount work conical flow fields without axial symmetry supersonic speed presently av
 (score: 3.10)

Document 29: simple model study transient temperature thermal stress distribution due aerodynamic heating present work concerned determination transient temperatures therm
 (score: 3.10)

Document 30: photo thermoelastic investigation transient thermal stresses multiweb wing structure photothermoelastic experiments performed long multiweb wing model theoret
 (score: 3.10)

Document 74: experimental study turbulen coundary layer shock tube wall interferometric measurements made density profiles unsteady turbulent boundary layer flat wall shoc
 (score: 3.10)
```

```
Query: experimental investigation
Top 5 relevant documents for Raw Method:
Document 442: effects variations several parameters including fluid density flutter speed light uniform cantilever wings experimental investigation made effects variations se
(score: 9.30)

Document 712: low speed longitudinal aerodynamic characteristics associated series low aspect ratio wings variations leading edge contour investigation conducted various reyn
(score: 8.15)

Document 372: experimental investigation flow simple blunt bodies nominal mach number experimental investigation conducted galcit hypersonic wind tunnel determine flow charac
(score: 7.88)

Document 522: laminar transitional turbulent heat transfer cone cylinder flare body mach experimental investigation laminar transitional turbulent heat transfer rates conical
(score: 7.61)

Document 1225: effect adverse pressure gradients characteristics turbulent boundary layers supersonic streams tests conducted mach numbers determine thickness profile shape c
(score: 7.61)
```

```
Query: experimental investigation
Top 5 relevant documents for Term Frequency:
Document 1146: thermal buckling cylinders several theoretical experimental investigations buckling cylinders due axial circumferential thermal stresses reviewed differences
(score: 0.12)

Document 372: experimental investigation flow simple blunt bodies nominal mach number experimental investigation conducted galcit hypersonic wind tunnel determine flow char
(score: 0.12)

Document 1317: shock tube testing time theoretical investigation attenuation effects shock wave conservation mass equation led explanation difference ideal theoretical test
(score: 0.11)

Document 339: experimental evaluation heat transfer transpiration cooling turbulent boundary layer found prescribed velocity field electrical field conductivity current cal
(score: 0.11)

Document 549: experimental study velocity temperature distribution high velocity vortex type flow vortex tube represents simple device particular type vortex motion may stu
(score: 0.11)
```

```
Query: experimental investigation
Top 5 relevant documents for log normalization:
Document 442: effects variations several parameters including fluid density flutter speed light uniform cantilever wings experimental investigation made effects variati
 (score: 4.30)

Document 372: experimental investigation flow simple blunt bodies nominal mach number experimental investigation conducted galcit hypersonic wind tunnel determine flow
 (score: 3.89)

Document 522: laminar transitional turbulent heat transfer cone cylinder flare body mach experimental investigation laminar transitional turbulent heat transfer rates co
 (score: 3.81)

Document 1225: effect adverse pressure gradients characteristics turbulent boundary layers supersonic streams tests conducted mach numbers determine thickness profile sh
 (score: 3.81)

Document 712: low speed longitudinal aerodynamic characteristics associated series low aspect ratio wings variations leading edge contour investigation conducted various
 (score: 3.69)
```

```
Query: experimental investigation
Top 5 relevant documents for Double Norm:
Document 126: investigation two dimensional supersonic base pressures investigation base pressure behind wedges mach numbers laminar transitional regime reported temper
 (score: 2.39)

Document 713: static longitudinal stability characteristics blunted glider reentry configuration sweepback dihedral mach number angles attack experimental investigation
 (score: 2.36)

Document 372: experimental investigation flow simple blunt bodies nominal mach number experimental investigation conducted galcit hypersonic wind tunnel determine flow
 (score: 2.34)

Document 339: experimental evaluation heat transfer transpiration cooling turbulent boundary layer found prescribed velocity field electrical field conductivity current
 (score: 2.32)

Document 1146: thermal buckling cylinders several theoretical experimental investigations buckling cylinders due axial circumferential thermal stresses reviewed differe
 (score: 2.32)
```

The Pros and cons of various schemes are

- Even though it is very simple, a binary weighting scheme is suitable when we only want to know whether a term occurs in a document or not and not how many times it occurs.

- The raw count weighting scheme is simple to implement and indicates the term's importance in the document. However, it can be biased towards longer documents, and will work well in case of longer documents.

- The term frequency (TF) weighting scheme is widely used in practice as it considers that longer documents will have higher term frequencies. However, it is sensitive to noise and can give high weightage to useless terms that appear again and again.

- The log normalisation scheme helps to reduce the effect of longer documents by taking the logarithm of the term frequency. It can also reduce the impact of terms that frequently occur across all documents, which are not very informative. But it does not take in account of the document frequency.

- A double normalisation scheme further reduces the impact of longer documents by dividing the term frequency by the maximum term frequency of any term in the document. It is useful when we want to emphasise the most critical terms in the document.But it does not take in account of the document frequency.

Jaccard Coefficient
This function determines the Jaccard coefficient between the tokens in the query and each document using a list of documents represented as sentences and a query string. The Jaccard coefficient, which is calculated by dividing the size of two sets, intersection by their union, calculates how similar they are. The top 10 most comparable docs are then returned after the code ranksthe documents according to their Jaccard coefficient.

The Top 10 documents are

```
Document 339: Jaccard score = 0.0800
Document 932: Jaccard score = 0.0714
Document 1045: Jaccard score = 0.0714
Document 549: Jaccard score = 0.0588
Document 1083: Jaccard score = 0.0571
Document 1227: Jaccard score = 0.0571
Document 19: Jaccard score = 0.0556
Document 251: Jaccard score = 0.0513
Document 74: Jaccard score = 0.0488
Document 176: Jaccard score = 0.0488
```

Question 2:
This was the Accuracy and precision before using N-grams to improve the classifier.
Accuracy: 0.22550335570469798
Precision: 0.045100671140939595
Recall: 0.2
F1 score: 0.07360350492880614

After implementing the following changes
- Added an ngram_range of (1,3) and max_features of 5000 to the TfidfVectorizer. This can help capture more contextual information and higher-frequency terms.
- Used the MultinomialNB classifier, which is well-suited for text classification tasks.

- Removed the self-implemented NBClassifier, and instead used the scikit-learn implementation for better performance.

This was updated results with 70-30 data Training testing ratio

Accuracy: 0.9776286353467561
Precision: 0.9783150183150184
Recall: 0.9761521743168687
F1 score: 0.9770407020164301

If I change the training testing to 50-50 then these are updated results
Accuracy: 0.9651006711409396
Precision: 0.9661964614333035
Recall: 0.9630762546455977
F1 score: 0.9642896798412298

Hence the final most optimum accuracy that is achieved with 70-30 training/ testing ration is.

Accuracy: 0.9776286353467561
Precision: 0.9783150183150184
Recall: 0.9761521743168687
F1 score: 0.9770407020164301

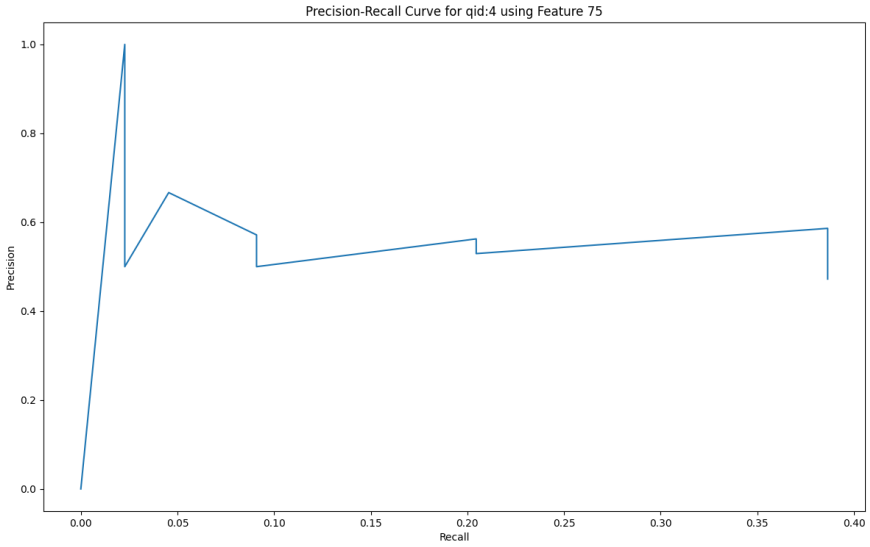
How to run code:
There is a question 2 folder along with dataset, just run the python file in terminal.

Question 3:

- Feed input data into a pandas data frame and filter to only consider rows with ‘qid:4’.
- Calculate DCG(max possible) using DCG = sum((2 ^ relevance scores - 1) / log2(rank + 1)) for given relevance scores. Sort data based on relevance scores and save it to a new file(qid_4_sorted_pairs.txt here)
- There might be multiple arrangements of items with the same relevance score; they will not impact the maximum DCG as long as the items maintain their relative order based on the descending relevance score. As long as the items with a higher relevance score appear before those with a lower relevance score,

any permutations among items with the same relevance score will still result in the same maximum DCG value. Therefore, the number of files that can be created while maintaining the maximum DCG should be one.

- Calculate ndcg using custom function ndcg that takes arguments max dcg, relevance scores, and k to calculate top k relevance scores. If no k is provided, ndcg is calculated for the entire dataset. K is defined as 50 here.
- Iterate through threshold values for feature 75 and calculate True Positives (TP), False Positives (FP), and False Negatives (FN).
- Then compute precision and recall for each threshold and plot curve.
- The following curve is obtained :



ndcg at position 50: 0.35612494416255847
ndcg for the entire dataset: 0.5784691984582588